

Ingesting data into Cloud Object Stores with RAZ Authorizations

Date published: 2019-12-16

Date modified: 2024-04-03

CLOUDERA

Legal Notice

© Cloudera Inc. 2024. All rights reserved.

The documentation is and contains Cloudera proprietary information protected by copyright and other intellectual property rights. No license under copyright or any other intellectual property right is granted herein.

Unless otherwise noted, scripts and sample code are licensed under the Apache License, Version 2.0.

Copyright information for Cloudera software may be found within the documentation accompanying each component in a particular release.

Cloudera software includes software from various open source or other third party projects, and may be released under the Apache Software License 2.0 (“ASLv2”), the Affero General Public License version 3 (AGPLv3), or other license terms. Other software included may be released under the terms of alternative open source licenses. Please review the license and notice files accompanying the software for additional licensing information.

Please visit the Cloudera software product page for more information on Cloudera software. For more information on Cloudera support services, please visit either the Support or Sales page. Feel free to contact us directly to discuss your specific needs.

Cloudera reserves the right to change any products at any time, and without notice. Cloudera assumes no responsibility nor liability arising from the use of products, except as expressly agreed to in writing by Cloudera.

Cloudera, Cloudera Altus, HUE, Impala, Cloudera Impala, and other Cloudera marks are registered or unregistered trademarks in the United States and other countries. All other trademarks are the property of their respective owners.

Disclaimer: EXCEPT AS EXPRESSLY PROVIDED IN A WRITTEN AGREEMENT WITH CLOUDERA, CLOUDERA DOES NOT MAKE NOR GIVE ANY REPRESENTATION, WARRANTY, NOR COVENANT OF ANY KIND, WHETHER EXPRESS OR IMPLIED, IN CONNECTION WITH CLOUDERA TECHNOLOGY OR RELATED SUPPORT PROVIDED IN CONNECTION THEREWITH. CLOUDERA DOES NOT WARRANT THAT CLOUDERA PRODUCTS NOR SOFTWARE WILL OPERATE UNINTERRUPTED NOR THAT IT WILL BE FREE FROM DEFECTS NOR ERRORS, THAT IT WILL PROTECT YOUR DATA FROM LOSS, CORRUPTION NOR UNAVAILABILITY, NOR THAT IT WILL MEET ALL OF CUSTOMER’S BUSINESS REQUIREMENTS. WITHOUT LIMITING THE FOREGOING, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, CLOUDERA EXPRESSLY DISCLAIMS ANY AND ALL IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, QUALITY, NON-INFRINGEMENT, TITLE, AND FITNESS FOR A PARTICULAR PURPOSE AND ANY REPRESENTATION, WARRANTY, OR COVENANT BASED ON COURSE OF DEALING OR USAGE IN TRADE.

Contents

Ingesting data into CDP Object Stores with RAZ authorization.....	4
Understand the use case.....	4
Meet the prerequisites.....	4
Build the data flow.....	5
Configure each object store processor.....	6
Set permissions in Ranger.....	12
Start the data flow.....	15
Verify data flow operation.....	15
Monitoring your data flow.....	16
Next steps.....	16

Ingesting data into CDP Object Stores with RAZ authorization

You can use an Apache NiFi data flow to ingest data into your object stores in a CDP Public Cloud environment that is enabled for Ranger Authorization Service (RAZ). Follow these steps:

Understand the use case

Learn how you can use object store processors to build an end-to-end flow that ingests data into Cloud object stores in a CDP environment that is enabled for fine-grained access control through RAZ.

The object store processors are:

- `ListCDPObjectStore`: Lists the FlowFiles in the source bucket that you want to ingest.
- `FetchCDPObjectStore`: Fetches the FlowFiles from the source bucket. If the files are compressed, this processor uncompresses the files.
- `PutCDPObjectStore`: Writes the contents of a FlowFile to the specified directory in the target bucket.
- `DeleteCDPObjectStore`: Deletes the ingested data from the source bucket.

These object store processors have the following advantages:

- The configuration of these processors is simple and easy.
- These processors can access the underlying object store of the cloud provider where the NiFi cluster is running. If you are running NiFi in Azure, these processors can be used to access ADLS. If you are running NiFi in AWS, these processors can be used to access S3. If you are running NiFi in Google Cloud, these processors can be used to access Google Cloud Storage.
- These processors are cloud-agnostic. If you have a multi-cloud architecture with, for example, a cluster running in AWS and another running in Azure, you can move this flow from one cloud provider to another with minimal change such as the input bucket.
- These processors are integrated with RAZ. With RAZ you can define policies in Ranger to specify who has access to what in the object store.
- Unlike HDFS processors that can only access the buckets that is configured for the data lake for your CDP environment, the object store processors can access any location in the underlying object store.

Meet the prerequisites

Use this checklist to make sure that you meet all the requirements before you start building your data flow.

- You have a CDP username and password set to access Data Hub clusters. The predefined resource role of this user is at least `EnvironmentUser`. This resource role provides the ability to view Data Hub clusters and set the `FreeIPA` password for the environment.
- Your user is synchronized to the CDP Public Cloud environment.
- You have a Flow Management Data Hub cluster running in your CDP Public Cloud environment.
- Your CDP user has been added to the appropriate pre-defined Ranger access policies to allow access to the NiFi UI.
- You have a source bucket on your object store from which you want to fetch data.
- You have created a target bucket on your object store for the processed data to be moved into.
- You have created a role with policies attached allowing read and write access to the object store you want to use in your data flow.



Note: To leverage the RAZ capabilities for Ranger defined policies that determine fine-grained access control to the object store, enable your CDP Public Cloud environment for RAZ.

If the environment is not RAZ enabled, the processors will not integrate with RAZ but will still work using IDBroker and the corresponding mappings.

Build the data flow

Learn how you can build a data flow using the object store processors to fetch data from one bucket, and after data transformation, ingest the data into another bucket. This involves opening Apache NiFi in your Flow Management cluster, adding processors and other data flow objects to your canvas, and connecting your data flow elements.

About this task

Use the `ListCDPObjectStore` and `FetchCDPObjectStore` processors to list and fetch data from your source bucket. Then, after transforming the data, use the `PutCDPObjectStore` to push the transformed data into the target bucket. Use the `DeleteCDPObjectStore` to delete your data in the source bucket.

Regardless of the type of flow you are building, the first steps in building your data flow are generally the same. Open NiFi, add your processors to the canvas, and connect the processors to create the flow.

Procedure

1. Open NiFi in Data Hub.
 - a) To access the NiFi service in your Flow Management Data Hub cluster, navigate to Management Console service Data Hub Clusters .
 - b) Click the tile representing the Flow Management Data Hub cluster you want to work with.
 - c) Click NiFi in the Services section of the cluster overview page to access the NiFi UI.

The screenshot displays the Cloudera Management Console interface for a cluster named 'cfm-nifi-cluster'. The left sidebar shows navigation options like Dashboard, Environments, Data Lakes, User Management, Data Hub Clusters, Data Warehouses, ML Workspaces, Classic Clusters, Shared Resources, and Global Settings. The main content area shows cluster details including status (Running), nodes (4), creation time (09/10/21, 02:24 PM GMT+2), and cluster template (7.2.11 - Flow Management Light Duty with Apache NiFi, Apache NiFi Registry). Below this, environment details for AWS are shown, including name (gvteticaden-pm-env-13-oregon), data lake (vett-pm-data-lake-13-oregon), credential (gvteticaden-pm-aws-prod-sandbox), region (us-west-2), and availability zone (us-west-2b). The Services section highlights NiFi, CM-UI, and NiFi Registry. Cloudera Manager Info shows CM URL, CM version (7.4.3), and runtime version (7.2.11-1.cdh7.2.11.p0.16654898). At the bottom, there are two tables: 'Management' and 'NiFi'. The 'Management' table lists a CM Server with ID 'i-06c0b1ceca1889c10', status 'Running', and FQDN 'cfm-nifi-cluster-management0.gvtetica.a465-9q4k.cloudera.site'. The 'NiFi' table lists a NiFi instance with ID 'i-099738af84c939fc6', status 'Running', and FQDN 'cfm-nifi-cluster-nifi1.gvtetica.a465-9q4k.cloudera.site'.

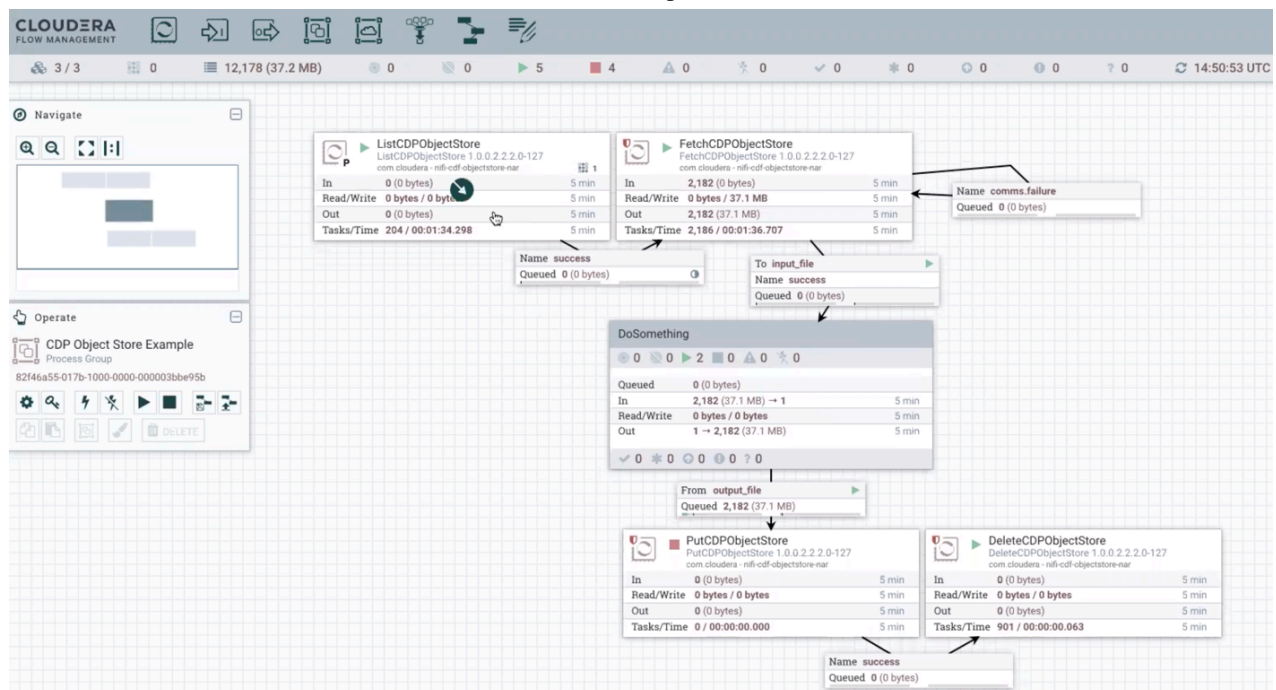
You will be logged into NiFi automatically with your CDP credentials.

2. Create the data flow by adding the `ListCDPObjectStore` processors onto the canvas.
 - a) Drag and drop the processor icon into the canvas.
In the dialog box you can choose which processor you want to add.
 - b) Select the processor of your choice from the list.
 - c) Click Add or double-click the required processor type to add it to the canvas.
3. Add the `FetchCDPObjectStore` processor.
4. Add any processors you require to transform your data.
5. Add the `PutCDPObjectStore` processor to write data to the target bucket.
6. Add the `DeleteCDPObjectStore` processor to write data to the target bucket.
7. Connect the processors to create the data flow by clicking the connection icon in the first processor, and dragging and dropping it on the second processor.

A Create Connection dialog appears with two tabs: Details and Settings. You can configure the connection's name, flowfile expiration time period, thresholds for back pressure, load balance strategy and prioritization.

Results

This is an example data flow with the `ListCDPObjectStore`, `FetchCDPObjectStore`, `PutCDPObjectStore`, and `DeleteCDPObjectStore` processors:



What to do next

Configure each object store processor in your data flow.

Configure each object store processor

Learn how you can configure the object store processors for your ingest data flow.

About this task

Configure the processor according to the behavior you expect in your data flow. Make sure that you set all required properties, as you cannot start the processor until all mandatory properties have been configured. Property values can be parameterized.

Procedure


1. Launch the Configure Processor window, by right clicking the `ListCDPObjectStore` processor and selecting Configure.
This gives you a configuration dialog with the following tabs: Settings, Scheduling, Properties, Comments.
2. Configure the `ListCDPObjectStore` processor properties and click Apply.

Figure 1: ListCDPObjectStore processor properties

The screenshot shows the 'Processor Details' window for the 'ListCDPObjectStore' processor. The 'Properties' tab is active, displaying a table of configuration properties. The status is 'Running' and there is a 'STOP & CONFIGURE' button in the top right. The table lists various properties such as Storage Location, Directory, Kerberos Credentials Service, CDP Username, CDP Password, Recurse Subdirectories, Record Writer, File Filter, File Filter Mode, Minimum File Age, and Maximum File Age.

Property	Value
Storage Location	#{input-bucket}
Directory	/\${now():toNumber():minus(8640000):format("yyyy-MM-...}
Kerberos Credentials Service	No value set
CDP Username	#{username}
CDP Password	Sensitive value set
Recurse Subdirectories	true
Record Writer	No value set
File Filter	[^].*
File Filter Mode	Directories and Files
Minimum File Age	No value set
Maximum File Age	No value set

Table 1: ListCDPObjectStore processor properties

Property	Description	Example value for ingest data flow
Storage Location	<p>The input bucket that contains the data that you want to ingest.</p> <p>You can create a parameter for the input bucket.</p> <p> Important: If you do not set the Storage Location property, the processor will set the storage location to the data lake.</p>	s3a://my-input-bucket-nifi
Directory	Path to the data files in the source bucket.	/\${now():toNumber():minus(8640000):format("yyy-MM-dd", "GMT")}
CDP Username	<p>Your CDP account user name.</p> <p>You can create a parameter for the CDP account credentials.</p>	srv_nifi-logs-ingest

Property	Description	Example value for ingest data flow
CDP Password	Your CDP account password. You can create a parameter for the CDP account credentials.	
Recurse Subdirectories	Set to true if you want to include subdirectories in the data ingest operation.	true

3. Configure the FetchCDPObjectStore processor properties and click Apply.

Figure 2: FetchCDPObjectStore processor properties

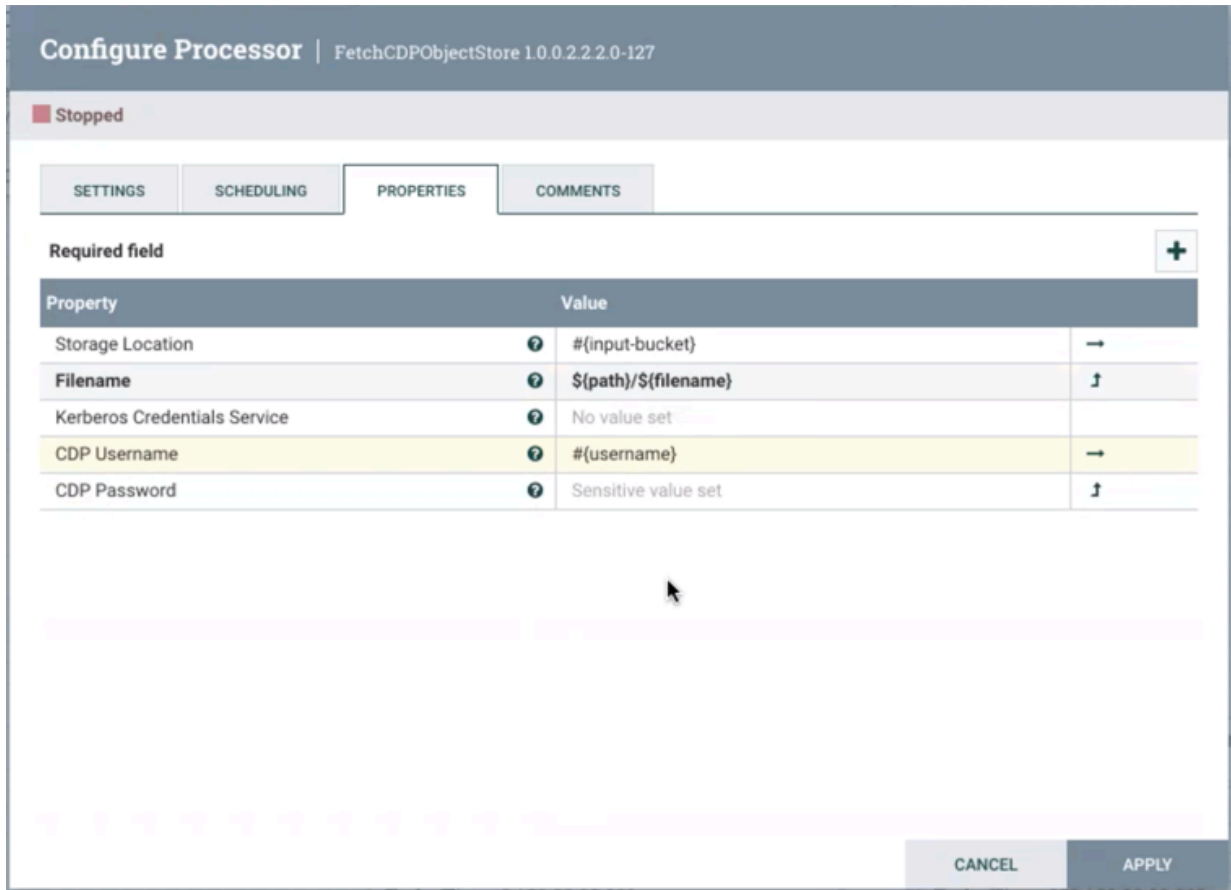



Table 2: FetchCDPObjectStore processor properties

Property	Description	Example value for ingest data flow
Storage Location	The input bucket that contains the data that you want to ingest. You can create a parameter for the input bucket and provide your workload account credentials to access the bucket.  Important: If you do not set the Storage Location property, the processor will set the storage location to the data lake.	s3a://my-input-bucket-nifi

Property	Description	Example value for ingest data flow
Filename	The fully-qualified filename of the file to fetch from the source bucket. You do not need to set this property because the default value will use the attributes of the flow files created by the ListCDPObjectStore processor.	\$(path)/\$(filename)
CDP Username	Your CDP account user name. You can create a parameter for the CDP account credentials.	srv_nifi-logs-ingest
CDP Password	Your CDP account password. You can create a parameter for the CDP account credentials.	

4. Configure the PutCDPObjectStore processor properties and click Apply.

Figure 3: PutCDPObjectStore processor properties

Configure Processor | PutCDPObjectStore 1.0.0.2.2.0-127

Stopped

SETTINGS SCHEDULING **PROPERTIES** COMMENTS

Required field +

Property	Value
Directory	/
Storage Location	#{output-bucket}
Conflict Resolution Strategy	fail
Kerberos Credentials Service	No value set
CDP Username	#{username}
CDP Password	Sensitive value set

CANCEL APPLY

Table 3: PutCDPObjectStore processor properties

Property	Description	Example value for ingest data flow
Directory	The directory to which files should be written.	/
Storage Location	The target bucket. You can create a parameter for the target bucket.	s3a://my-output-bucket-nifi

Property	Description	Example value for ingest data flow
Conflict Resolution Strategy	Indicates what should happen when a file with the same name already exists in the target directory.	You can enter one of the following values: <ul style="list-style-type: none">• fail• ignore• replace
CDP Username	Your CDP account user name. You can create a parameter for the CDP account credentials.	srv_nifi-logs-ingest
CDP Password	Your CDP account password. You can create a parameter for the CDP account credentials.	

- Configure the DeleteCDPObjectStore processor properties and click Apply.

Figure 4: DeleteCDPObjectStore processor properties

Processor Details

Running (1) STOP & CONFIGURE


SETTINGS SCHEDULING **PROPERTIES** COMMENTS

Required field

Property	Value
Storage Location	#{input-bucket}
Path	\${path}/\${filename}.gz
Kerberos Credentials Service	No value set
CDP Username	#{username}
CDP Password	Sensitive value set

OK

Table 4: DeleteCDPObjectStore processor properties

Property	Description	Example value for ingest data flow
Storage Location	<p>The source bucket that contains the data that you have ingested and now want to delete.</p> <p>You can create a parameter for the input bucket.</p> <p> Important: If you do not set the Storage Location property, the processor will set the storage location to the data lake.</p>	s3a://my-input-bucket-nifi
Path	Path to the data files in the source bucket that you want to delete.	\$(path)/\$(filename).gz
CDP Username	<p>Your CDP account user name.</p> <p>You can create a parameter for the CDP account credentials.</p>	srv_nifi-logs-ingest
CDP Password	<p>Your CDP account password.</p> <p>You can create a parameter for the CDP account credentials.</p>	

What to do next

Create policies in Ranger to enable the CDP user to access the source and target buckets.

Set permissions in Ranger

Create custom Ranger policies to enable the CDP user to read and write to the source and target buckets.

About this task

From your RAZ-enabled environment, access the Ranger service for your cloud provider. Then create a policy give the CDP user read and write access to the source bucket. Create another policy to give the CDP user read and write access to the target bucket.

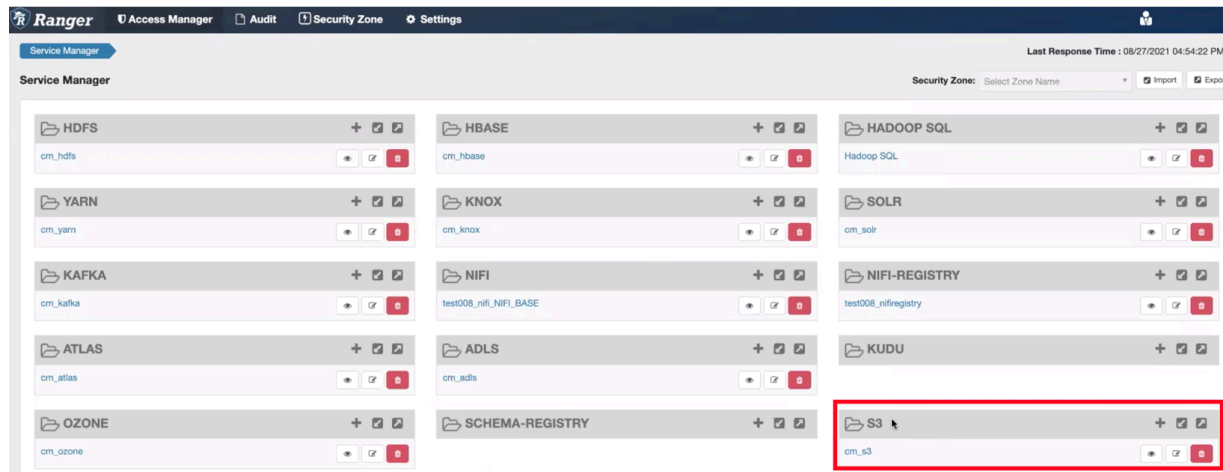
Procedure

1. From your Data Hub cluster, select Ranger from the list of services and log into Ranger.

The Ranger **Service Manager** page displays.

2. Select your cluster from your Cloud provider service folder.

The following image shows an Amazon S3 service folder with a Data Hub cluster.



The **List of Policies** page appears.

3. Click **Add New Policy**.

The **Create Policy** page appears.

4. Add the following details to allow the user to access the source bucket:
 - a) Enter a unique name for the policy. For example, Logs input.
 - b) Specify your source bucket name. For example, s3a://my-input-bucket-nifi.
 - c) In the Path field, specify the path to a specific directory or file. Or, to indicate any path, enter /.
 - d) In the Allow Condition section, specify the CDP user name in the Select User field. For example, srv_nifi-log s-ingest.
 - e) In the Permissions fields, enter read and write.
 - f) Click Add to save the policy.

Policy Details ✕

Service Name : cm_s3

Service Type : s3

Policy Details :

Policy Type	Access
Policy ID	97
Version	4
Policy Name	Logs input Normal Enabled
Policy Labels	--
S3 Bucket	my-input-bucket-nifi
Path	/ recursive
Description	--
Audit Logging	Yes

Allow Condition :

Select Role	Select Group	Select User	Permissions	Delegate Admin
--	--	srv_nifi-logs-ingest	read write	<input type="checkbox"/>

Exclude from Allow Conditions :

Select Role	Select Group	Select User	Permissions	Delegate Admin
No Data Found !!				

Deny All Other Accesses : FALSE

5. Add the following details to allow the user to access the target bucket:
 - a) Enter a unique name for the policy. For example, Logs output.
 - b) Specify your target bucket name. For example, s3a://my-output-bucket-nifi.
 - c) In the Path field, specify the path to a specific directory or file. Or, to indicate any path, enter /.
 - d) In the Allow Condition section, specify the CDP user name in the Select User field. For example, srv_nifi-log s-ingest.
 - e) In the Permissions fields, enter read and write.
 - f) Click Add to save the policy.

Policy Details ✕

Service Name : cm_s3

Service Type : s3

Policy Details :

Policy Type	Access
Policy ID	98
Version	1
Policy Name	Logs output Normal Enabled
Policy Labels	--
S3 Bucket	my-output-bucket-nifi
Path	/ recursive
Description	--
Audit Logging	Yes

Allow Condition :

Select Role	Select Group	Select User	Permissions	Delegate Admin
--	--	srv_nifi-logs-ingest	read write	<input type="checkbox"/>

Exclude from Allow Conditions :

Select Role	Select Group	Select User	Permissions	Delegate Admin
No Data Found !!				

Deny All Other Accesses : **FALSE**

Results

When you start the data flow, the processors using the CDP user credentials can list and fetch from the source bucket and put and delete in the target bucket.

What to do next

Start the data flow.

Start the data flow

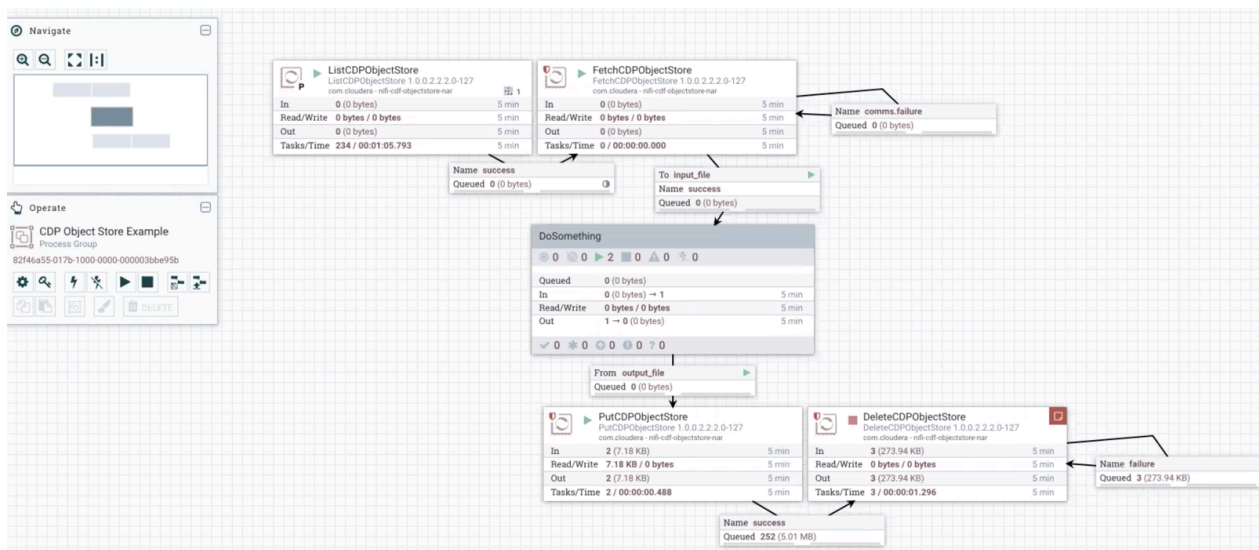
When your flow is ready, you can begin ingesting data into your object store. Learn how to start your object store ingest data flow.

Procedure

1. To initiate your data flow, select all the data flow components you want to start.
2. Click the Start icon in the Actions toolbar.

Alternatively, right-click a single component and choose Start from the context menu. Data should be read from the input bucket and it should be written to the defined folder of your object store.

Results



What to do next

It is useful to check that data is running through the flow you have created.

Verify data flow operation

Learn how you can verify the operation of your object store ingest data flow.

About this task

There are a number of ways you can check that data is running through the flow you have built.

Procedure

1. You can verify that NiFi processors are not producing errors.

2. You can look at the processors in the UI, where you can see the amount of data that has gone through them. You can also right click on the processors, or on connections to view status history.
3. You can check that the data generated appears in your target bucket. To do this, return to the dedicated folder of your object store bucket, where you should see your files listed.
You may have to refresh the page depending on your browser/settings.

Monitoring your data flow

Learn about the different monitoring options for your object store ingest data flow in CDP Public Cloud.

You can monitor your data flow for information about health, status, and details about the operation of processors and connections. NiFi records and indexes data provenance information, so you can conduct troubleshooting in real time.

Data statistics are rolled up on a summary screen (the little table icon on the top right toolbar which lists all the processors). You can use the `MonitorActivity` processor to alert you, if for example you have not received any data in your flow for a specified amount of time.

If you are worried about data being queued up, you can check how much data is currently queued. Process groups also conveniently show the totals for any queues within them. This can often indicate if there is a bottleneck in your flow somewhere, and how far the data has got through that pipeline.

Another option to check that data has fully passed through your flow is to check out data provenance to see the full history of your data.

Next steps

Learn about the different options that you have after building a simple object store ingest data flow in CDP Public Cloud.

Moving data to the cloud is one of the cornerstones of any cloud migration. Cloud environments offer numerous deployment options and services. This example data flow provides you with a model to design more complex data flows for moving and processing data as part of cloud migration efforts.

You can build a combination of on-premise and public cloud data storage. You can use this solution as a path to migrate your entire data to the cloud over time—eventually transitioning to a fully cloud-native solution or to extend your existing on-premise storage infrastructure, for example for a disaster recovery scenario. Cloud storage can provide secure, durable, and extremely low-cost options for data archiving and long-term backup for on-premise datasets.

You can also use cloud services without storing your data in the cloud. In this case you would continue to use your legacy on-premise data storage infrastructure, and work with on-demand, cloud-based services for data transformation, processing and analytics with the best performance, reliability and cost efficiency for your needs. This way you can manage demand peaks, provide higher processing power, and sophisticated tools without the need to permanently invest in computer hardware.